## AMENDMENT OF THE CLAIMS

1. (Currently Amended) A method for logging a successful authentication of a user, the method comprising:

generating, by an embedded system, a session identification in response to the authentication to identify a login session for the user;

allocating, by the embedded system, application independent memory to create a memory location of the embedded system for shared access by more than one applications of the embedded system;

arbitrating the shared access to at least a portion of the memory location of the embedded system by checking a semaphore in the memory location to determine whether an application is currently accessing the at least a portion of the memory location and, if not, granting access to the at least a portion of the memory location by storing an in-use indication in the semaphore, the in-use indication comprising the session identification,

~~storing the session identification in the memory location of the embedded system in response to receipt of unshared access to at least a portion of the memory location~~, the memory location being configured to retain the session identification independent of de-allocations of memory for the more than one applications, to authenticate the user for the login session; and by releasing the at least a portion of the memory location for access by other applications by resetting the in-use indication in the semaphore ~~the unshared access to the at least the portion of the memory location in response to storing the session identification~~; and

transmitting the session identification from the embedded system to a computer from which the user accessed the embedded system.

2. (Original) The method of claim 1, further comprising storing additional session information with the session identification to associate the additional session information with the login session.

3. (Original) The method of claim 1, further comprising:
associating a time indication with the session identification and
removing the session identification from the memory location after a period of inactivity by the user.

4. (Original) The method of claim 1, further comprising locking the memory location while accessing an entry for the session identification in the memory location.

5. (Original) The method of claim 1, wherein generating a session identification comprises generating a random number, the random number uniquely identifying the user's login session.

6. (Previously Presented) The method of claim 1, wherein storing the session identification comprises storing the session identification in a shared memory buffer.

7. (Original) The method of claim 1, wherein transmitting the session identification comprises generating a cookie and transmitting the cookie to a web browser utilized by the user.

8. (Previously Presented) The method of claim 1, further comprising authenticating the login session for the user in response to receipt of the session identification associated with the user with a request, wherein authenticating the user comprises comparing the session identification with session identifications previously stored in the memory location.

9. (Currently Amended) An apparatus of an embedded system for logging a successful authentication of a user for the embedded system, the apparatus comprising:

a memory location of the embedded system to retain a session identification independent of de-allocation of memory for more than one applications executed on the embedded system;

an arbitrator to facilitate shared access to the memory location by the more than one applications executed on the embedded system and to prevent simultaneous access to the memory location by more than one of the applications executed on the embedded system by checking a semaphore in the memory location to determine whether an application is currently accessing the memory location and, if not, granting access to at least a portion of the memory location by storing an in-use indication in the semaphore and by releasing the at least a portion of the memory location for access by other of the more than one applications by resetting the in-use indication in the semaphore ~~granting unshared access to one of the more than one applications at a time~~;

a session creator coupled with the memory location to generate the session identification in response to the authentication to identify a login session of the user, store the session identification in the memory location in response to ~~a grant of unshared~~ the granting access to the at least a portion of the memory location by the arbitrator, and transmit the session identification to the user's computer; and

a session authenticator coupled with the memory location to authenticate the user for an application of the individual applications executed on the server in response to receipt of the session identification upon verifying that the session identification is stored in the memory location.

10. (Previously Presented) The apparatus of claim 9, wherein the arbitrator locks at least one entry in the memory location while accessing the at least one entry for the session identification in the memory location and unlocks the at least one entry of the memory location in response to completion of access of the at least one entry of the memory location.

11. (Original) The apparatus of claim 9, wherein the memory location comprises at least a portion of a shared memory buffer.

12. (Original) The apparatus of claim 9, wherein the session creator comprises a session identification generator to generate a unique, random number to associate with the user and the login session.

13. (Original) The apparatus of claim 9, wherein the session creator comprises a bundle generator to bundle the session identification with additional session information, associating the additional session information with the login session.

14. (Original) The apparatus of claim 13, wherein the bundle generator comprises a time stamper to associate a time indication with the session identification and the session authenticator comprises a time monitor coupled with the memory location to remove the session identification from the memory location after a period of inactivity by the user.

15. (Previously Presented) The apparatus of claim 9, wherein the session creator comprises a bundle transmitter to transmit a bundle having the session identification to a web browser, wherein the web browser is utilized by the user to communicate with the embedded system.

16. (Original) The apparatus of claim 9, wherein the session authenticator comprises an identification comparator to compare the session identification received from the user with session identifications previously stored in the memory location.

17. (Currently Amended) A computer program product for logging successful authentication of a user for an embedded system, the computer program product comprising:

a computer usable storage medium having computer usable program code embodied therewith, the computer usable program code comprising:

computer usable program code configured to perform operations, comprising:

generating, by an embedded system, a session identification in response to the authentication to identify a login session for the user;

allocating, by the embedded system, application independent memory to create a memory location of the embedded system for shared access by more than one applications of the embedded system;

arbitrating the shared access to at least a portion of the memory location of the embedded system by checking a semaphore in the memory location to determine whether an application is currently accessing the at least a portion of the memory location and, if not, granting access to the at least a portion of the memory location by storing an in-use indication in the semaphore, the in-use indication comprising the session identification,

~~storing the session identification in the memory location of the embedded system in response to receipt of unshared access to at least a portion of the memory location~~, the memory location being configured to retain the session identification independent of de-allocations of memory for the more than one applications, to authenticate the user for the login session; and by releasing the at least a portion of the memory location for access by other applications by resetting the in-use indication in the semaphore ~~the unshared access to the at least the portion of the memory location in response to storing the session identification~~; and

transmitting the session identification from the embedded system to a computer-from which the user accessed the embedded system.

18. (Currently Amended) The computer program product of claim 17, wherein the operations further comprise arbitrating access to the memory location to grant ~~the unshared~~ access to prevent simultaneous access by more than one of the applications ~~to the entry comprising the session identifier~~.

19. (Previously Presented) The computer program product of claim 17, wherein the operations further comprise associating a time indication with the session identification to facilitate removal of the session identification from the memory location upon expiration of the login session.

20. (Previously Presented) The computer program product of claim 17, wherein the operations further comprise locking the memory location while accessing the memory location.

21. (Previously Presented) The computer program product of claim 17, wherein generating a session identification comprises generating a unique, random number to associate with the user and the login session.

22. (Previously Presented) The computer program product of claim 17, wherein authenticating the user comprises receiving the session identification from the user and comparing the session identification with session identifications in the memory location.